Delegating Private Quantum Computations

Anne Broadbent

Department of Mathematics and Statistics
University of Ottawa
abroadbe@uottawa.ca

Abstract

We give a protocol for the delegation of quantum computation on encrypted data. More specifically, we show that in a client-server scenario, where the client holds the encryption key for an encrypted quantum register held by the server, it is possible for the server to perform a universal set of quantum gates on the quantum data. All Clifford group gates are non-interactive, while the remaining non-Clifford group gate that we implement (the $\pi/8$ gate) requires the client to prepare and send a single random auxiliary qubit (chosen among four possibilities), and exchange classical communication. This construction improves on previous work, which requires either multiple auxiliary qubits or two-way quantum communication. Using a reduction to an entanglement-based protocol, we show privacy against any adversarial server according to a simulation-based security definition.

1 Introduction

Today's computing paradigm displays seemingly contradictory requirements. On one hand, computations are often delegated to remote powerful computing centers, while on the other hand, the data that is being processed is expected to remain private. We thus face the conundrum of wanting to compute on encrypted data. One specific scenario that allows data to be encrypted by one party and processed by another is known as *fully homomorphic encryption* [15, 20].

This paper[1] addresses the problem of performing *quantum* computations on encrypted *quantum* data. In one way, we relax the requirements of fully homomorphic encryption by allowing *interaction*, but at the same time, we strengthen the requirements by asking for information-theoretic security. This is an asymmetric scenario—it deals with a quantum server (or quantum *cloud* architecture), a particularly relevant scenario due to the current challenges in building quantum computational devices. This scenario is also considered in [1, 8, 9]. We show that an almost-classical client can delegate the execution of any quantum computation to a remote quantum server, and that this computation can be performed on quantum data that is encrypted via the quantum one-time pad [2]; informally, privacy is maintained since the server *never* learns the encryption key. An important requirement of any protocol for delegated computation on encrypted data is that the operations performed by the client should be significantly easier to perform than the computation itself. In our scenario, we achieve this since the client does not require the capacity of universal quantum

[1] This paper was initially submitted in its current form (up to minor corrections, updates and formatting) to the Proceedings of the 32nd International Cryptology Conference (CRYPTO 2012), where it was rejected. The results were then improved to include an experimental demonstration and eventually appeared as [14]. Thus this version appears here for the first time in print and will be of special interest to those wishing to focus on the theory contribution in [14].

computation. She only requires the ability to perform encryption and decryption (for this she needs to be able to apply single-qubit Pauli operators); she also must be able to prepare send random qubits chosen from a set of four possibilities. This set of states is unitarily equivalent to the set $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$, which are known as the BB84 states, for the crucial role that they play in the quantum key distribution protocol known by the same name [6]. Such a client does not require quantum memory and can be implemented with current technology, for instance, using photon polarization [5, 17]. We suppose that the client is honest and prove security against any cheating server via simulations. Similar functionality has been achieved before [1, 8, 9].

Compared to prior work, our contribution has the advantage of providing a conceptually simple proof of correctness, together with a security definition and proof that is applicable to all types of prior information, including shared entanglement. Additionally, our protocol is more efficient in terms of quantum and classical communication. Compared to [8], our gain for a general quantum computation is by a constant factor; nevertheless, this means that, using current technology [4], our protocol could lead to the experimental delegation of a wider class of private quantum computations.

A sample application of our protocol would be the delegated, private execution of Shor's algorithm [22] which can be used to factor in polynomial time on a quantum computer (this computation is widely believed to be intractable on a classical computer). Since the computation is performed on an encrypted input, the server will not know which integer he is factoring; if this integer corresponds to an RSA public key [21] then the server will not know which public key he is helping to break. We thus see that quantum computing on encrypted data is useful for the delegation of problems that can be solved in quantum polynomial time, with the underlying assumption that they cannot be solved in classical polynomial time.

However, applications of delegating private quantum computations are foreseeable *even if* it turns out that quantum computers are no more powerful than classical ones, since delegated computation on encrypted *quantum* data is also achieved. This could be useful, for instance, to enable a client (with no universal quantum computer) to perform quantum circuits on quantum data such as *quantum money* [13] or *quantum coins* [18].

2 Contributions and Related Work

In order to achieve our results, we consider the scenario of quantum computing on encrypted data: one party holds a quantum register while the other party holds the encryption key. It is well known that performing a Clifford group circuit (or, more generally, a stabilizer circuit) on quantum data encrypted with the one-time pad can be achieved non-interactively: the server (holding the encrypted data) applies the target gates, while the client (holding secret encryption keys) simply adjusts her knowledge of the encryption key (see Sections 4.1 and 4.2). What remains in order to perform a universal quantum computation is to show how to implement a non-Clifford group gate on encrypted data.

Our main contribution is a simple protocol for computing a $\pi/8$ gate over encrypted data (Figure 12). We define security via simulation and show that the final protocol is secure against any malicious server (Section 4.4). We use as proof technique the method of transforming a qubit-based protocol into an equivalent protocol that is more easily proved secure, but that involves entanglement. This technique is attributed to Shor and Preskill [23], who used it in the context of proving the security of the BB84 [6] quantum key exchange protocol, and has since appeared in the context of quantum message authentication [3] and cryptography in the bounded-quantum-storage model [11].

We emphasize that the protocol achieves the same level of privacy as the quantum one-time pad, which is the highest possible level of security: it depends only on the correctness of quantum mechanics and in particular does not rely on any computational assumptions. In contrast, fully homomorphic encryption [15] provides computational security only because it uses a public-key encryption scheme.

We have phrased our contribution in terms of performing a *publicly-known* circuit on encrypted data. Hiding the entire computation is possible simply by executing a universal circuit on an encrypted input, part of which contains the description of the target circuit to be implemented. Furthermore, the protocol can easily be adapted to allow the server to provide an input.

Previous results achieve similar (or even identical) functionality, with an similar level of security, but require more resources:

1. The *secure assisted quantum computation* protocol of Childs [9], accomplishes the same functionality as our protocol, but with a significant difference in that the protocol involves two-way quantum communication and the client needs to be able to execute a two-qubit *swap* gate. We give more details on how our protocol differs from Childs' in Section 4.3.

2. The protocol for *universal blind quantum computation* of Broadbent, Kashefi and Fitzsimmons [8] achieves a similar functionality, by using more resources. While the goal of blind quantum computing is first and foremost to hide the computation itself, the protocol also achieves computation on encrypted data. Because our protocol does not require that the circuit be hidden, we manage to reduce the requirements in terms of communication: while [8] requires for each gate (including the identity), 24 bits of forward communication, eight bits of backward communication and eight auxiliary qubits, our protocol reduces the communication to null for all but the execution of a non-Clifford group gate; in the specific case of the $\pi/8$ gate, the interaction consists of a single auxiliary qubit and two classical bits (one bit in each direction). Furthermore, [8] requires that auxiliary qubits be prepared from a set of eight possible states, while we manage to reduce this to four (see Section 4.3). Note that the universal blind quantum computation protocol was recently experimentally demonstrated [4].

3. The protocol for *quantum prover interactive proofs* of Aharonov, Ben-Or and Eban [1] establishes, on top of the functionality that we implement, a *verification* mechanism to ensure that the server is performing the correct computation. The cost of this construction is that the client needs to prepare auxiliary quantum systems of size polynomial in the parameter determining the security of the verification. Our protocol does not provide any verification mechanism, but manages to significantly limit the quantum power needed by the client.

Finally, our work is related to the more general scenario of two-party quantum computation, where it is the case that *both* parties hold keys to the encrypted quantum data. Dupuis, Nielsen and Salvail [12] gave a protocol for two-party secure quantum computation in the case of *specious* (a version of quantum semi-honest) adversaries. Certain of our sub-protocols display a similarity (local Clifford group gates are essentially identical). However, our contribution for the R-gate protocol is very different since it requires no quantum interaction. In this respect, the work of [12] is closer to the work of [9].

3 Preliminaries

We assume the reader is familiar with the basics of quantum information [19]. Recall the following notation: $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, the Pauli gates $\mathsf{X} : |j\rangle \mapsto |j \oplus 1\rangle$ and $\mathsf{Z} : |j\rangle \mapsto (-1)^j|j\rangle$, as well as the single-qubit Hadamard and phase gates, $\mathsf{H} : |j\rangle \mapsto \frac{1}{\sqrt{2}}(|0\rangle + (-1)^j|1\rangle)$, $\mathsf{P} : |j\rangle \mapsto (i)^j|j\rangle$. Recall also the two-qubit gate $\mathsf{CNOT} : |j\rangle|k\rangle \mapsto |j\rangle|j \oplus k\rangle$. An *EPR-pair* is a pair of maximally entangled qubits, $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.

3.1 Quantum circuits and circuit identities

The *Clifford Group* [16] is the set of operators that conjugate Pauli operators into Pauli operators. A universal gate set for *Clifford group circuits* consists of the Pauli gates themselves, together with H, P and CNOT. *Stabilizer circuits* are formed by adding the operations of single-qubit measurements and auxiliary qubit preparation to the Clifford group circuits. Stabilizer circuits are not universal for quantum computation [16], however supplementing with *any* additional gate outside of the group (such as $\mathsf{R} : |j\rangle \mapsto e^{ij\pi/4}|j\rangle$ or the Toffoli gate $\mathsf{T} : |j\rangle|k\rangle|\ell\rangle \mapsto |j\rangle|k\rangle|\ell \oplus jk\rangle$) is necessary and sufficient for universality. We interchangeably refer to the R-gate as the $\pi/8$ gate.

We will make use of the following identities which all hold up to an irrelevant global phase: $\mathsf{XZ} = \mathsf{ZX}$, $\mathsf{PZ} = \mathsf{ZP}$, $\mathsf{PX} = \mathsf{XZP}$, $\mathsf{RZ} = \mathsf{ZR}$, $\mathsf{RX} = \mathsf{XZPR}$, $\mathsf{P}^2 = \mathsf{Z}$ and $\mathsf{P}^{a\oplus b} = \mathsf{Z}^{a\cdot b}\mathsf{P}^{a+b}$ (for $a, b \in \{0,1\}$).

In order to derive and prove our results, we make use of known techniques for manipulating quantum circuits. Of significant relevance to our work are the techniques developed by Childs, Leung, and Nielsen [10] to manipulate circuits that produce an output that is correct *up to known Pauli corrections*. These techniques are based on a variant of teleportation introduced by Zhou, Leung, and Chuang [25] (see Figure 1, as well as Appendix A for a derivation of this circuit identity). Here and in the following figures, measurements are performed in the computational basis.

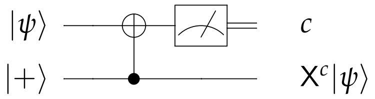


Figure 1: X-teleportation circuit [25].

We also use the fact that P and Z commute with control (Figure 2).

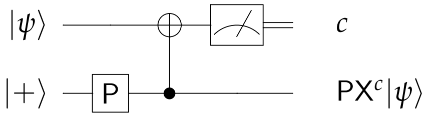


Figure 2: Circuit identity: the P-gate commutes with control. A similar identity holds if we replace the P-gate with a Z-gate.

Finally, we make use of an entanglement-based circuit that prepares a qubit $\mathsf{Z}^d\mathsf{P}^y|+\rangle$ for uniformly random bits y and d (Figure 3). Correctness of this circuit is easy to verify.

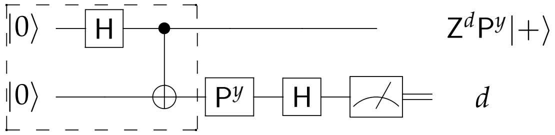


Figure 3: Circuit identity: entanglement-based circuit that prepares a qubit $\mathsf{Z}^d\mathsf{P}^y|+\rangle$ for uniformly random bits y and d (here, y is chosen uniformly at random, and d is determined by the measurement). The circuit in the dashed box prepares an EPR-pair.

3.2 Classical and quantum encryption

The classical *one-time pad* is an encryption procedure that maps each bit j of a plaintext to $j \oplus r$ for a uniformly random key bit r (which we denote $r \in_R \{0,1\}$). Since the ciphertext $j \oplus r$ is uniformly random (as long as r is unknown), the plaintext j is perfectly concealed. The *quantum one-time pad* [2] is the quantum analog of the classical one-time pad. The encryption procedure for a single qubit consists of uniformly randomly applying an operator in $\{\mathsf{I}, \mathsf{X}, \mathsf{Z}, \mathsf{XZ}\}$, or equivalently, applying $\mathsf{X}^a\mathsf{Z}^b$ for uniformly random bits a and b (here, I is the identity). This maps any single qubit to the maximally mixed state on one qubit, which we denote $\mathbb{1}_2$; thus the quantum plaintext is perfectly concealed.

3.3 Quantum registers and channels

A *quantum register* is a collection of qubits in some finite dimensional Hilbert space, say $\mathcal{X}$. We denote $\mathrm{D}(\mathcal{X})$ the set of density operators acting on $\mathcal{X}$. The set of all linear mappings from $\mathcal{X}$ to $\mathcal{Y}$ is denoted by $\mathrm{L}(\mathcal{X}, \mathcal{Y})$, with $\mathrm{L}(\mathcal{X})$ being a shorthand for $\mathrm{L}(\mathcal{X}, \mathcal{X})$. A linear super-operator $\Phi : \mathrm{L}(\mathcal{X}) \to \mathrm{L}(\mathcal{Y})$ is *admissible* if it is completely positive and trace-preserving. Admissible super-operators represent mappings from density operators to density operators, that is, they represent the most general quantum maps.

Given admissible super-operators Φ and Ψ that agree on input space $\mathrm{L}(\mathcal{X})$ and output space $\mathrm{L}(\mathcal{Y})$, we are interested (for cryptographic purposes) in characterizing how "indistinguishable" these processes are. The *diamond norm* provides such a measure: given that Φ or Ψ is applied with equal probability, the optimal procedure to determine the identity of the channel with only one use succeeds with probability $1/2 + \|\Phi - \Psi\|_\diamond/4$. Here, $\|\Phi - \Psi\|_\diamond = \max\{\|(\Phi \otimes \mathbb{1}_\mathcal{W})(\rho) - (\Psi \otimes \mathbb{1}_\mathcal{W})(\rho)\|_1 : \rho \in \mathrm{D}(\mathcal{X} \otimes \mathcal{W})\}$, where $\mathcal{W}$ is any space with dimension equal to that of $\mathcal{X}$ and $\mathbb{1}_\mathcal{W}$ is the identity in $\mathrm{L}(\mathcal{W})$, and where the *trace norm* of an operator X is defined as $\|X\|_1 = \mathrm{Tr}\sqrt{X^* X}$.

4 Delegating private quantum computations

A general quantum circuit can be decomposed into a sequence of the following: gates in $\{\mathsf{X}, \mathsf{Z}, \mathsf{H}, \mathsf{P}, \mathsf{CNOT}, \mathsf{R}\}$, auxiliary qubit preparation in $|0\rangle$ and single-qubit computational basis measurements (strictly speaking, this set is redundant; the choice of these gates will become clear later). We show in the following sections that these operations can be executed by a server who has access only to the input in its encrypted form (where the encryption is the quantum one-time pad), and we show that the output can nevertheless be decrypted by the client, and that the server does not learn anything about the input. In order to accomplish this, we give a series of protocols, each accomplishing the execution of a circuit element. For each such protocol, the client (who knows the encryption key for the input to the protocol) can compute a decryption key that, if applied to the output of the protocol, would result in the output of the circuit element applied to the unencrypted input.

Sections 4.1 and 4.2 show protocols for stabilizer circuit elements, while Section 4.3 gives a protocol for a non-Clifford group gate (this is the only gate that uses interaction). In all cases, we give explicit constructions for pure states; it is straightforward to verify that the same constructions work on systems that are entangled.

We can see each of the protocols as gadgets that implement a circuit element, up to a known re-interpretation of the key. In order to execute a larger target circuit, each of its circuit elements is executed in sequence as one of these gadgets; it is sufficient for the client to re-adjust her knowledge of the encryption keys on each relevant quantum wire after each gadget. Our protocol for quantum computing on encrypted data is given as:

1. The client encrypts her register with the quantum one-time pad and sends the encrypted register to the server.

2. The client and server perform the gadgets as given in Section 4.1–4.3, according to the circuit that is to be executed, with the client re-adjusting the encryption keys on each relevant quantum wire after each gadget.

3. The server returns the output register to the client, who decrypts it according to the key that she has computed.

Note that this high-level protocol does not involve any interaction other than the sending (Step 1) and receiving (Step 3) of the encrypted data. Only a *single* gadget in the implementation of Step 2 in our construction is interactive (Section 4.3), and the quantum part of this interaction is one-way (from the client to the server), consisting of the sending of a single random auxiliary qubit in $\{\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle), \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle), \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)\}$, which can be sent, without loss of generality, at the beginning of the protocol. Thus our protocol for quantum computing on encrypted data is interactive, but completely classical, except for the initial sending of auxiliary random single-qubit states, as well as for the sending of the input and output registers (if they are quantum). Furthermore, a target circuit that implements only Clifford group operations can be executed with no interaction at all (except for the interaction in Steps 1 and 3). We now proceed to give the protocols (Sections 4.1, 4.2 and 4.3). Security is formally defined via simulation and proved in Section 4.4.

4.1 Protocols for measurement and auxiliary qubit preparation

The protocol for measuring a single qubit in the computational basis is given in Figure 4: the server simply performs a measurement on the encrypted qubit. The corresponding wire thus goes from a quantum wire (with encryption operation $\mathsf{X}^a\mathsf{Z}^b$) to a classical wire (with encryption key a); the client can easily take this into account.

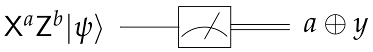


Figure 4: Protocol for measurement. Here, y denotes the outcome of the measurement on the unencrypted input $|\psi\rangle$.

As represented in Figure 5, the server may prepare an unencrypted auxiliary qubit in the $|0\rangle$ state and incorporate it into the computation. The client simply sets the encryption key for this qubit to be 0.

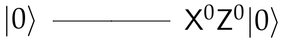


Figure 5: Protocol for auxiliary qubit preparation.

4.2 Protocols for Clifford group gates

A series of well-known relationships between the Pauli matrices and Clifford group operations [16] is the basis for the protocols given in Figures 6–10. Specifically, since the X and Z gates commute or anti-commute (and since we can safely ignore a global phase), Figures 6 and 7 can easily be seen as implementing a protocol for the X and Z-gates. Similarly, the relation $\mathsf{HX} = \mathsf{ZH}$ is sufficient to show the H-gate protocol given in Figure 8, and the facts that $\mathsf{PZ} = \mathsf{ZP}$ and $\mathsf{PX} = -i\mathsf{ZXP}$ show the P-gate protocol given in Figure 9. Finally, the protocol for the CNOT-gate as given in Figure 10 can be verified in a similar way.

Strictly speaking, in order to achieve universality, we do not need all of the protocols given above: once we have the protocol for the R-gate (given in Section 4.3 below), it is sufficient to combine it with the protocols for CNOT and H for universality. This can be seen since $\mathsf{P} = \mathsf{R}^2$, $\mathsf{Z} = \mathsf{P}^2$ and $\mathsf{X} = \mathsf{HZH}$. However, each of these decompositions requires at least two R-gates, and as we will see below, the protocol for an R-gate is relatively expensive (it uses an auxiliary qubit and classical interaction). It can thus be preferable to decompose a circuit into the redundant gate set that we have used as this reduces the cost of many gates. Also, by giving explicit protocols for all of these circuit elements, we have established that a stabilizer circuit can be performed on encrypted data *without any* interaction whatsoever, except for the exchanging of the encrypted data register.

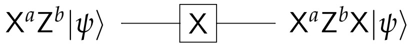

$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
X
$\mathsf{X}^a\mathsf{Z}^b\mathsf{X}|\psi\rangle$

Figure 6: Protocol for an X-gate.

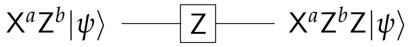

$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
Z
$\mathsf{X}^a\mathsf{Z}^b\mathsf{Z}|\psi\rangle$

Figure 7: Protocol for a Z-gate.

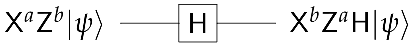

$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
H
$\mathsf{X}^b\mathsf{Z}^a\mathsf{H}|\psi\rangle$

Figure 8: Protocol for an H-gate.

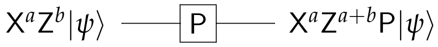

$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
P
$\mathsf{X}^a\mathsf{Z}^{a+b}\mathsf{P}|\psi\rangle$

Figure 9: Protocol for a P-gate.

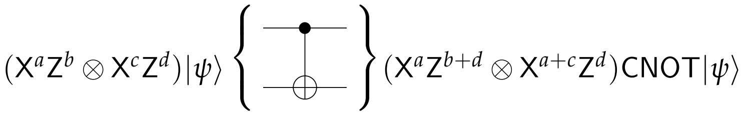

$(\mathsf{X}^a\mathsf{Z}^b \otimes \mathsf{X}^c\mathsf{Z}^d)|\psi\rangle$
$(\mathsf{X}^a\mathsf{Z}^{b+d} \otimes \mathsf{X}^{a+c}\mathsf{Z}^d)\mathsf{CNOT}|\psi\rangle$

Figure 10: Protocol for a CNOT-gate. Here, $|\psi\rangle$ is a two-qubit system.

4.3 Protocol for a non-Clifford group gate

The only remaining gate required to implement universal quantum computation is a non-Clifford group gate. We choose the R-gate.

Our first attempt at a protocol for an R-gate (Figure 11) follows the protocols given in the previous section: the server simply applies the R-gate to the encrypted data. However, this does not immediately work, since $\mathsf{RX} = \mathsf{XZPR}$ and so in the case that an X-encryption is present, the output picks up an undesirable P gate (this cannot be corrected by applying Pauli corrections). In [9], Childs arrives at this same conclusion, and then makes the astute observation that, in the case where $a = 1$, the server could be made to *correct* this erroneous P-gate by executing a correction (which consists of ZP). As long as the server does not find out if this correction is being executed or not, security holds.

This is where our approach takes a significantly different route compared to [9] or even [12]: while these references solve this problem with *two-way* quantum communication, we solve it with classical interaction

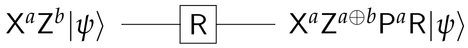

$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
R
$\mathsf{X}^a\mathsf{Z}^{a\oplus b}\mathsf{P}^a\mathsf{R}|\psi\rangle$

Figure 11: First attempt at a protocol for an R-gate; output requires a P correction if $a = 1$.

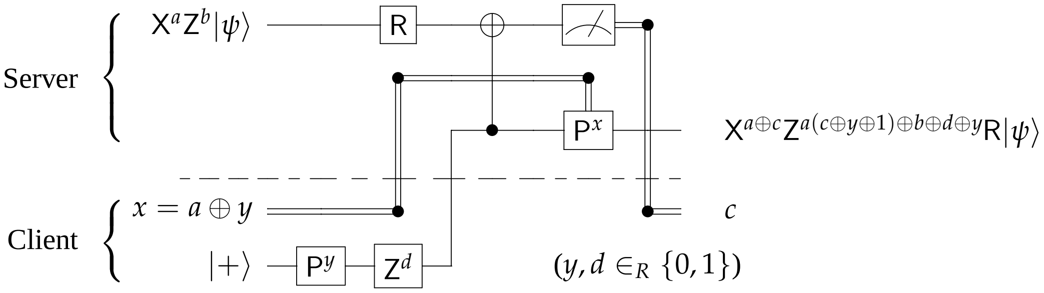

Server
$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
R
P^x
$\mathsf{X}^{a\oplus c}\mathsf{Z}^{a(c\oplus y\oplus 1)\oplus b\oplus d\oplus y}\mathsf{R}|\psi\rangle$
Client
$x = a \oplus y$
c
$|+\rangle$
P^y
Z^d
$(y, d \in_R \{0,1\})$

Figure 12: Protocol for an R-gate.

and a single forward auxiliary qubit randomly chosen out of four possibilities. See Figure 12, as well as the proof of correctness in Appendix A.

Compared to [8], we manage to halve the size of the set from which random qubits are chosen (from eight to four). This can be seen as due to the fact that [8] directly implements a hidden R-gate, while in Figure 12, the server first applies the R-gate, and applies a correction by performing a hidden P-gate, requiring less resources than a hidden R-gate.

4.4 Correctness and Security

Given that the correctness of each gadget has been shown, correctness of the main protocol is obvious: after each gadget, the client adapts her knowledge of the keys used to encrypt the system according to Figure 5–12; each gadget itself is correct, so the entire protocol implements the quantum circuit as desired.

Our protocol provides the same level of security as the one-time pad (Section 3), that is, it provides perfect (information-theoretic) privacy. The rest of this section formalizes the definition of privacy based on simulations and gives a proof based on the technique of giving an equivalent, entanglement-based protocol (see Section 2). For our definition of privacy, we have used notions similar to those introduced by Watrous in the context of quantum zero-knowledge interactive proof systems [24].

Formally, a protocol for delegated computation is specified by a pair (C, S) representing an honest client and an honest server (without loss of generality, both parties are quantum). As the client is always honest, the security property concerns interactions between pairs (C, S') where S' deviates arbitrarily from S. At the onset of the protocol, both parties agree on the classical input q which determines the general quantum circuit to be executed as an ordered series of gates acting on specified wires. The structure of the interaction between C and S is thus determined by q. At the same time, a quantum input $\rho_{\text{in}} \in \mathrm{D}(\mathcal{C} \otimes \mathcal{S})$ is distributed, C receiving the register in $\mathcal{C}$ and S receiving the register in $\mathcal{S}$. A cheating server S' is any quantum computational process that interacts with C according to the message structure determined by q. By allowing S' access to the input register $\mathcal{S}$, we explicitly allow S' to share prior entanglement with C's input; this also models any *prior* knowledge of S' and formalizes the notion that the protocol cannot be used to *increase* knowledge.

Let $\mathcal{Z}$ denote the output space of S' and let $\Phi_q : \mathrm{L}(\mathcal{S} \otimes \mathcal{C}) \to \mathrm{L}(\mathcal{Z})$ be the mapping induced by the interaction of S' with C. Security is defined in terms of the existence of a *simulator* $\mathscr{S}_{S'}$ for a given server S', which is a general quantum circuit that agrees with S' on the input and output dimensions. Such a simulator does not interact with C, but simply induces a mapping $\Psi_q : \mathrm{L}(\mathcal{S} \otimes \mathcal{C}) \to \mathrm{L}(\mathcal{Z})$ given by $\mathscr{S}_{S'}(\mathrm{tr}_{\mathcal{C}}\rho_{\mathrm{in}})$ on each input q. Informally, (C, S) is private if the two mappings, Φ_q and Ψ_q are indistinguishable for every choice of q and every choice of ρ_{in}. Allowing for an ϵ amount of leakage, we formalize this as the following: A protocol (C, S) for a delegated quantum computation is *ϵ-private* if for every server S' there exists a simulator $\mathscr{S}_{S'}$ such that for every classical input q,$\|\Phi_q - \Psi_q\|_\diamond \leq \epsilon$, where Φ_q is the mapping induced by the interaction of S' with the client C on input q and Ψ_q is the mapping induced by $\mathscr{S}_{S'}$ on input q.

Taking $\epsilon = 0$ gives the strongest possible security against a malicious server: it does not allow for even an ϵ amount of leakage, and allows the server to deviate arbitrarily (without imposing any computational bounds). This is the level of security that we claim for our protocol for delegated quantum computation: it is ϵ-private, with $\epsilon = 0$. The proof follows (although we do not formalize this notion here, we note that our proof method provides a simulator with the often-desirable property that it runs with essentially the same computational resources as the deviating server).

Fix a value for q. We construct a simulator $\mathscr{S}_{S'}$ by giving instructions how to prepare messages that replace the messages that the client C would send to the server S' in the real protocol. Privacy follows since we will show that these transmissions are identical to those in the real protocol.

A high-level sketch of the proof is that we modify the behaviour of the client in the main protocol (**Protocol 1**) in a way that the effect of the protocol is unchanged (meaning that both the output of the protocol and the view of the server is unchanged), yet the client delays introducing her input into the protocol until after her interaction with the server has ended (this makes the simulation almost trivial). In order to do so, we describe below an entanglement-based protocol (**Protocol 2**) as well as a delayed-measurement protocol (**Protocol 3**).

We first consider **Protocol 2**, which is an entanglement-based version of **Protocol 1**. In **Protocol 2**, we modify how the client prepares her messages, without modifying the server's actions or the effect of the protocol. Thus, the preparing and sending of an encrypted quantum register in step 1 of **Protocol 1** is replaced by an equivalent teleportation-based protocol, as given in Figure 13. Also, the R-gate protocol in step 2 of **Protocol 1** is replaced by an equivalent protocol as given in Figure 15. The protocol of Figure 15 can be seen to be correct via an intermediate protocol (Figure 14), in which the classical bit x from the client to the server becomes a uniformly random bit; this transformation is possible because in the protocol of Figure 12, $x = a \oplus y$ with y a random bit. Then choosing x to be random and $y = a \oplus x$ gives an equivalent protocol. The final entanglement-based protocol of Figure 15 is seen to be correct via the circuit identity given in Figure 3. The remaining protocols for stabilizer circuit elements are non-interactive and thus unchanged in **Protocol 2**.

The main advantage of considering **Protocol 2** instead of **Protocol 1** is that we can delay all the client's measurements (in Figures 13 and 15) until the output register is returned in step 3 of **Protocol 1**, without affecting the computation or the server's view of the protocol (because actions on different subsystems commute); call the result **Protocol 3**. In this delayed-measurement protocol, the messages from the client to the server can be chosen *before* any interaction with the server, and are thus clearly independent of the actions of S'.

Thus we construct a simulator $\mathscr{S}_{S'}$ that plays the role of the client in **Protocol 3**, *but that never performs any measurements* (thus, access to the actual input is not required). By the argument above, $\mathscr{S}_{S'}$ actually prepares the same transmissions as would C in **Protocol 1** interacting with S' on any input ρ_{in}. It follows that simulating S' on these transmissions will induce the same mapping as S' in the real protocol, and thus $\|\Phi_q - \Psi_q\|_\diamond = 0$.

Note that a malicious server may not necessarily follow the protocol, thus possibly interfering with the computation. Our protocol does not guard against this; detecting a cheating server can be done using a

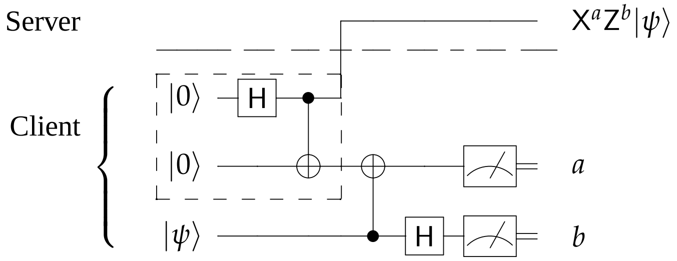


Figure 13: Protocol to encrypt and send a qubit using teleportation[7]. The circuit in the dashed box prepares an EPR-pair.

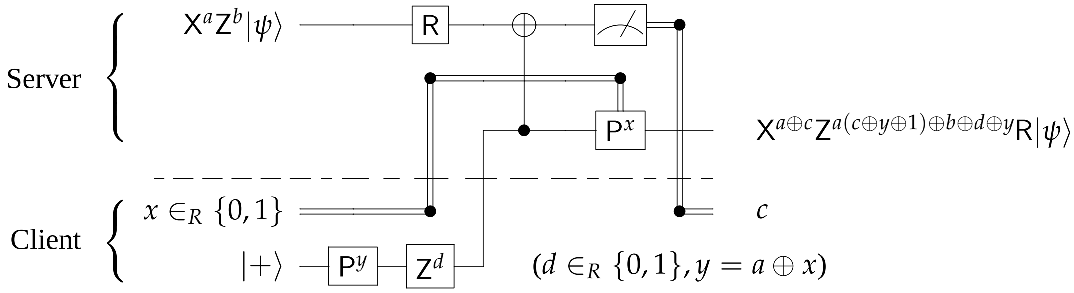

$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
R
Server
P^x
$\mathsf{X}^{a\oplus c}\mathsf{Z}^{a(c\oplus y\oplus 1)\oplus b\oplus d\oplus y}\mathsf{R}|\psi\rangle$
Client
$x \in_R \{0,1\}$
c
$|+\rangle$
P^y
Z^d
$(d \in_R \{0,1\}, y = a \oplus x)$

Figure 14: Intermediate Protocol for an R-gate. Compared to Figure 12, the classical message from the client to the server is chosen uniformly at random. This protocol performs the same computation as the protocol in Figure 12.

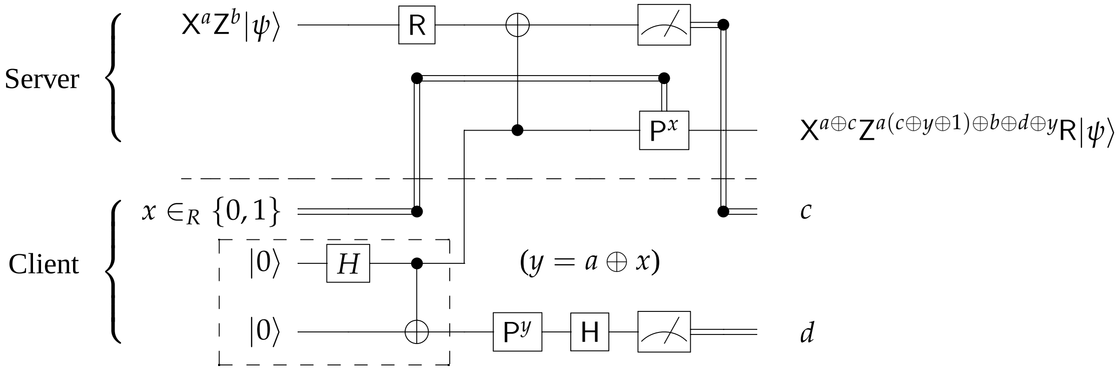

$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
R
Server
P^x
$\mathsf{X}^{a\oplus c}\mathsf{Z}^{a(c\oplus y\oplus 1)\oplus b\oplus d\oplus y}\mathsf{R}|\psi\rangle$
$x \in_R \{0,1\}$
c
Client
$|0\rangle$
H
$(y = a \oplus x)$
$|0\rangle$
P^y
H
d

Figure 15: Entanglement-based protocol for an R-gate. This protocol performs the same computation as the protocols in Figures 12 and 14. The circuit in the dashed box prepares an EPR-pair.

quantum authentication code [3], as is done in [1] (albeit by requiring the client to have more quantum power). It is important to note however that our privacy definition and proof holds against such a malicious server.

Acknowledgements

I would like to thank Richard Cleve, Serge Fehr, Stacey Jeffery and Debbie Leung for related discussions. This work was completed while the author was affiliated with the Canadian Institute for Advanced Research, the University of Waterloo and the Institute for Quantum Computing.

References

[1] D. Aharonov, M. Ben-Or, and E. Eban. Interactive proofs for quantum computations. In *Proceeding of Innovations in Computer Science 2010 (ICS 2010)*, pages 453–469, 2010.

[2] A. Ambainis, M. Mosca, A. Tapp, and R. De Wolf. Private quantum channels. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science (FOCS 2000)*, pages 547–553, 2000.

[3] H. Barnum, C. Crépeau, D. Gottesman, A. Smith, and A. Tapp. Authentication of quantum messages. In *Proceedings of the 43rd Annual IEEE Symposium on the Foundations of Computer Science (FOCS '02)*, pages 449–458, 2002.

[4] S. Barz, E. Kashefi, A. Broadbent, J. F. Fitzsimons, A. Zeilinger, and P. Walther. Demonstration of blind quantum computing. *Science*, 20:303–308, 2012.

[5] C. H. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin. Experimental quantum cryptography. *Journal of Cryptology*, 5:3–28, 1992.

[6] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179, 1984.

[7] C .H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, and W. Wootters. Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels. *Physical Review Letters*, pages 1895–1899, 1993.

[8] A. Broadbent, J. Fitzsimons, and E. Kashefi. Universal blind quantum computation. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2009)*, pages 517–526, 2009.

[9] A. Childs. Secure assisted quantum computation. *Quantum Information and Computation*, 5:456–466, 2005.

[10] A. M. Childs, D. W. Leung, and M. A. Nielsen. Unified derivations of measurement-based schemes for quantum computation. *Physical Review A*, 71:032318, 2005.

[11] I. B. Damgård, S. Fehr, L. Salvail, and C. Schaffner. Cryptography in the bounded-quantum-storage model. In *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science (FOCS 2005)*, 2005. Full version in SIAM Journal on Computing 37:1865-1890, 2008.

[12] F. Dupuis, J. B. Nielsen, and L. Salvail. Secure two-party quantum evaluation of unitaries against specious adversaries. In *Proceedings of the 30th International Cryptology Conference (CRYPTO 2010)*, pages 685–706, 2010. Detailed version available as `arXiv:1009.2096`.

[13] E. Farhi, D. Gosset, A. Hassidim, A. Lutomirski, and P. Shor. Quantum money from knots. In *3rd Innovations in Theoretical Computer Science (ITCS 2012)*, 2012. Available as `arXiv:1004.5127`.

[14] K. A. G. Fisher, A. Broadbent, L .K. Shalm, Z. Yan, J. Lavoie, R. Prevedel, T. Jennewein, and K .J. Resch. Quantum computing on encrypted data. *Nature communications*, 5, 2014.

[15] C. Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the 41st annual ACM symposium on Theory of Computing (STOC '09)*, pages 169–178, 2009.

[16] D. Gottesman. The Heisenberg representation of quantum computers. In *Group 22: Proceedings of the XXII International Colloquium on Group Theoretical Methods in Physics*, pages 32–43, 1998.

[17] P. Kok, W. J. Munro, K. Nemoto, T. C. Ralph, J. P. Dowling, and G. J. Milburn. Linear optical quantum computing with photonic qubits. *Review of Modern Physics*, 79:135–174, 2007.

[18] M. Mosca and D. Stebila. Quantum coins. *Error-Correcting Codes, Finite Geometries and Cryptography*, 523:35–47, 2010.

[19] M .A. Nielsen and I .L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2000.

[20] R. Rivest, L. Adleman, and M. Dertouzos. On data banks and privacy homomorphisms. In *Foundations of Secure Computation*, pages 169–177, 1978.

[21] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21:120–126, 1978.

[22] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Scientific and Statistical Computing*, 26:1484–1509, 1997.

[23] P. W. Shor and J. Preskill. Simple proof of security of the BB84 quantum key distribution protocol. *Physical Review Letters*, 85:441–444, 2000.

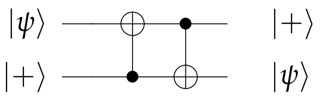
[24] J. Watrous. Zero-knowledge against quantum attacks. *SIAM Journal on Computing*, 39:25–58, 2009. Preliminary version in *Proceedings of the 38th ACM Symposium on Theory of Computing (STOC '06)*, pages 296–305, 2006.

[25] X. Zhou, D. W. Leung, and I. L. Chuang. Methodology for quantum logic gate construction. *Physical Review A*, 62:052316, 2000.

A Correctness of the R-gate protocol

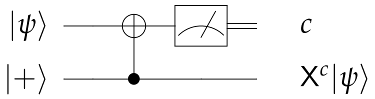
We give below a step-by-step proof of the correctness of the R-gate protocol as given in Figure 12. The basic building block is the circuit identity for an X-teleportation from [25], which we re-derive here.

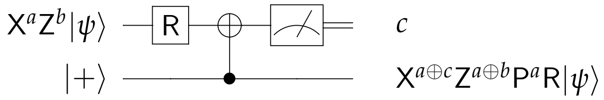
1. Our first circuit identity swaps a qubit $|\psi\rangle$ with the state $|+\rangle$ and is easy to verify.



2. We can measure the top qubit in the above circuit and classically control the output correction. We have thus re-derived the circuit corresponding to the "X-teleportation" of [25].

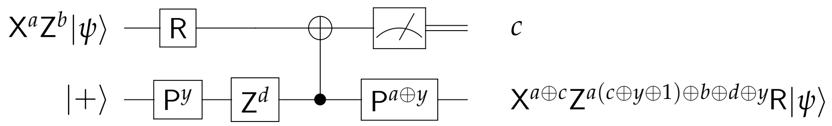
3. Next, we re-define the input to be $\mathsf{R}\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$, so the output becomes $\mathsf{X}^c\mathsf{R}\mathsf{X}^a\mathsf{Z}^b|\psi\rangle = \mathsf{X}^{a\oplus c}\mathsf{Z}^{a\oplus b}\mathsf{P}^a\mathsf{R}|\psi\rangle$.




$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
$|+\rangle$
R
c
$\mathsf{X}^{a\oplus c}\mathsf{Z}^{a\oplus b}\mathsf{P}^a\mathsf{R}|\psi\rangle$

4. Then add three gates (P^y, Z^d, $\mathsf{P}^{a\oplus y}$) to the bottom wire (see circuit below). Applying identities from Section 3, we get as output what we expect:

$$\begin{aligned}
&\mathsf{P}^{a\oplus y}\mathsf{Z}^d\mathsf{P}^y\mathsf{X}^{a\oplus c}\mathsf{Z}^{a\oplus b}\mathsf{P}^a\mathsf{R}|\psi\rangle\\
&= \mathsf{Z}^{a\cdot y}\mathsf{P}^{a+y}\mathsf{Z}^d\mathsf{P}^y\mathsf{X}^{a\oplus c}\mathsf{Z}^{a\oplus b}\mathsf{P}^a\mathsf{R}|\psi\rangle\\
&= \mathsf{Z}^{d\oplus a\cdot y\oplus y}\mathsf{P}^a\mathsf{X}^{a\oplus c}\mathsf{Z}^{a\oplus b}\mathsf{P}^a\mathsf{R}|\psi\rangle\\
&= \mathsf{Z}^{d\oplus a\cdot y\oplus y}\mathsf{X}^{a\oplus c}\mathsf{Z}^{a(a\oplus c)}\mathsf{P}^a\mathsf{Z}^{a\oplus b}\mathsf{P}^a\mathsf{R}|\psi\rangle\\
&= \mathsf{X}^{a\oplus c}\mathsf{Z}^{d\oplus a\cdot y\oplus y\oplus a^2\oplus a\cdot c}\mathsf{Z}^b\mathsf{R}|\psi\rangle\\
&= \mathsf{X}^{a\oplus c}\mathsf{Z}^{a(c\oplus y\oplus 1)\oplus b\oplus d\oplus y}\mathsf{R}|\psi\rangle
\end{aligned}$$


$\mathsf{X}^a\mathsf{Z}^b|\psi\rangle$
$|+\rangle$
R
P^y
Z^d
$\mathsf{P}^{a\oplus y}$
c
$\mathsf{X}^{a\oplus c}\mathsf{Z}^{a(c\oplus y\oplus 1)\oplus b\oplus d\oplus y}\mathsf{R}|\psi\rangle$